工程数学学报稿件编号：
（由编辑编写）

公共交通系统最佳路径模型与算法

汤志高，王继利，曹莹瑛
指导教师：曹华林
（海军航空工程学院（青岛），青岛 266041）

编者按: 这是一篇从各方面看都很不错的论文。用乘客出行时常考虑的四个指标，分别构成字典序来选择出行路线，所建立的两个模型、相应的算法及数据结构都比较恰当。六个实例的计算准确无误，第三问建立的规划模型和算法基本可行，但将一个地铁站周围可以换乘的几个公汽站视为一点的做法欠妥。

摘 要: 本文将题中的直行、环行线路信息分别抽象，以$MATLAB$软件中元胞数组为载体储存直达数据库Q，针对用户不同需求（换乘次数、总耗时、总费用、转站车辆是否始发）构成不同非负有向赋权图和相应的权矩阵，建立多目标线性规划模型求解。为使算法的空间、时间复杂度相对平衡，在求解换乘次数小于等于2次的时候采用线路相交算法[5]来计算可行方案，换乘次数大于2次时采用邻接算法或线性规划法求解。最后把各目标下求得的出行路线方案集以字典序方式输出至用户终端。

关键词: 非负有向赋权图 邻接算法 字典序
分类号: **AMS**(2000) 90C11 **中图分类号**:0221 **文献标识码**: A

1 问题重述（略）

2 模型假设

1）不存在重名站点，且只有同名站点允许换乘（利用地铁站除外）；
2）各路径发车频度相同，并且不考虑交班时段；
3）环行线路双向行驶，在车过终点后不另行收费；
4）始发站、终点站出入地铁站需要步行4分钟；
5）地铁站周围任意公汽站点间步行耗时2分钟；
6）用户对步行时间与乘车时间等同看待。

3 问题分析

3.1 不同公汽线路抽象

根据题中原始数据信息，将线路分三类（上下行经过站点相同、环行线、上下行经过站点不同），下面对这三种线路进行数据处理：

1）上下行经过站点相同

这种线路虽然上下行经过站点相同，但线路的方向不同，下行线和上行线需要抽象为两条有向线路：

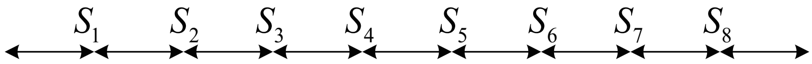

Figure 1: 上下行经过站点相同抽象示意图

2）环行线

本文假设环形路线为双环（双向行驶），所以在对这两条线路进行抽象时，为保证任意两站点距离最近，需要把每条线路抽象为4条有向线路：$\overrightarrow{ABCD}$、$\overrightarrow{ADCB}$、$\overrightarrow{CDAB}$、$\overrightarrow{CBAD}$

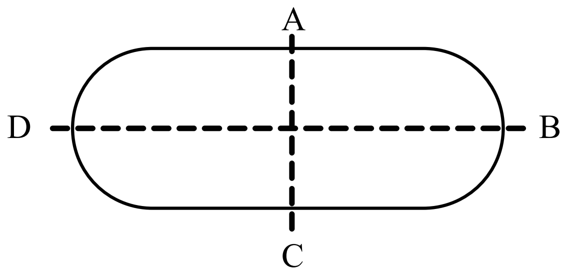

Figure 2: 环行线抽象示意图

3）上下行线经过站点不同

由于下行线与上行线经过站点不同，该种线路显然需要抽象成两条有向线路处理：

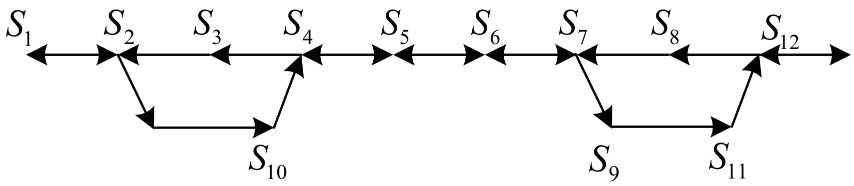

Figure 3: 不同过站上下行路线抽象示意图

3.2 “直达数据库Q”的建立

<table>
<tr><td>Cell{1,1}</td><td>Cell{1,2}
<table><tr><th>车号</th><th>费用</th><th>耗时</th></tr><tr><td>L001</td><td>2</td><td>27</td></tr><tr><td>L076</td><td>3</td><td>18</td></tr></table></td><td>Cell{1,3}</td></tr>
<tr><td>Cell{2,1}</td><td>Cell{2,2}</td><td>Cell{2,3}</td></tr>
</table>

Figure 4: 元胞结构示意图

上图中Cell{1, 2}代表Q中第1行第2个元胞（即从站点S0001到站点S0002的直达公交线路信息），元胞中队列的每一行代表一辆直达车信息。

3.3 最少换乘次数的确定

1）直达线路数矩阵

统计Q中各站点间直达线路数可以建立$A=(a_{ij})_{N\times N}$，N表示站点数目，矩阵元素a_{ij}表示第i个站点到第j个站点直达线路数n，其中，当$i=j$时为无效路径，设定值为0，即：

$$a_{ij}=\begin{cases} n & (i\neq j) \\ 0 & (i=j) \end{cases}$$

2）换乘线路数矩阵

直达矩阵**A**为$N\times N$阶方阵，矩阵的2次幂中元素表示任两站点间通过1次转乘的路线数$A^2=A\cdot A$；以A^n表示方阵**A**的n次幂，A_{kj}为站点$k\rightarrow j$的直达线路数，则：

$$A_{ij}^n=\sum_{k=1}^{N}A_{ik}^{n-1}\cdot A_{kj}\qquad (i,j=1...N;n>1)$$

3）最少换乘次数矩阵

矩阵$B=(b_{ij})_{N\times N}$表示从站点$i\rightarrow j$必要的最少换乘次数

$$b_{ij}=\begin{cases} \min\quad \left\{n|A_{ij}^n\neq 0,n\in N^*\right\}-1 & (i\neq j) \\ 0 & (i=j) \end{cases}$$

基于最少换乘次数矩阵B，可确定题中给出的6条实例线路最少换乘次数分别为1、2、1、1、2、1次，若同时考虑公汽与地铁第六线改变为0次。

4 模型I的分析与建立

4.1 权矩阵定义

引用图论相关知识，将题中所提供的公汽网络抽象成一个非负有向赋权图$\overrightarrow{G}=(V,E,W)$,$\overrightarrow{G}$ 中的每个顶点为每个不同的站点, 如果从$\overrightarrow{G}$中的顶点v_i到v_j 有直达路线，那么这两点之间存在有向边, 记做$(i,j)\in E$ ，方向为从i 指向j，相应的有$w(v_i,v_j)$ 称为该有向边的权，这样公汽网络就抽象为一个有向赋权图。赋权图中的权可根据不同的目标进行定义，分别定义为：

时间：$W^t=(t_{ij})_{n\times n}$，其分量 $t_{ij}=\begin{cases} t_{(v_i,v_j)} & \text{站点}v_i\text{至站点}v_j\text{的直达时间} \\ +\infty & \text{无直达线路} \end{cases}$

费用：$W^p=(p_{ij})_{n\times n}$，其分量$p_{ij}=\begin{cases} p_{(v_i,v_j)} & \text{站点}v_i\text{至站点}v_j\text{的直达费用} \\ +\infty & \text{无直达线路} \end{cases}$

始发：$W^f=(f_{ij})_{n\times n}$，其分量$f_{ij}=\begin{cases} f_{(v_i,v_j)} & \text{站点}v_i\text{至站点}v_j\text{的直达线路是否在}v_i\text{始发} \\ +\infty & \text{无直达线路} \end{cases}$

4.2 目标分析

目标一：换乘次数最少

基于4.1节有向赋权图，引入0-1决策变量x_{ij}表示弧(i,j) 是否在起点s与终点e的路上，则：

$$x_{ij}=\begin{cases}1 & 弧(i,j)位于v_s\rightarrow v_e的路上\\ 0 & ELSE\end{cases}$$

换乘次数最小可表示为：

$$Min \quad \sum_{(i,j)\in E} x_{ij}-1 \tag{1}$$

目标二：行程总时间最短

行程总时间=始发等待时间+乘车总时间+公汽换乘时间：

$$Min \quad 3+\sum_{(i,j)\in E} t_{ij}x_{ij}+\ 5\left(\sum_{(i,j)\in E} x_{ij}-1\right) \tag{2}$$

目标三：行程总费用最少

统计Q，建立q_{ij}表示$i\rightarrow j$车辆属性，$q_{ij}=\begin{cases}1 & 表示单一票制1元\\ 2 & 分段计价\end{cases}$

统计Q，建立s_{ij}表示$i\rightarrow j$过站数目，那么$i\rightarrow j$直达费用表示为：

$$p_{ij}=\begin{cases}1 & q_{ij}=1\\ 1 & q_{ij}=2,s_{ij}\in[1,20]\\ 2 & q_{ij}=2,s_{ij}\in[21,40]\\ 3 & q_{ij}=2,s_{ij}\in[41,+\infty]\end{cases}$$

行程总费用最少：

$$Min \quad \sum_{(i,j)\in E} p_{ij}x_{ij} \tag{3}$$

目标四：转乘车辆始发最多

$$Max \quad \sum_{(i,j)\in E} f_{ij}x_{ij} \tag{4}$$

4.3 约束分析

约束一：换乘次数约束

基于对目标一的分析，可得在有向赋权图中换乘次数表达式，以c 表示乘客所能接受的最大换乘次数，换乘次数约束可表示为：

$$\sum_{(i,j)\in E} x_{ij}-1\le c \qquad (c\in N^*) \tag{5}$$

约束二：最短路起讫点约束

在图$\overrightarrow{G}$中路过点分为“起点”、“中间点”、“终点”三类，起点只有出边而无入边，中间点既有入也有出边，终点只有入边。

$$\sum_{\substack{j=1 \\ (i,j)\in E}}^{n} x_{ij} - \sum_{\substack{j=1 \\ (i,j)\in E}}^{n} x_{ji} = \begin{cases} 1 & (i=s) \\ -1 & (i=e) \\ 0 & (i\neq s,e) \end{cases} \tag{6}$$

4.4 模型I的建立

基于本节4.1∼4.3分析，以式(1)∼(4)为目标，式(5)、(6)为约束，建立线性规划如下：

$$Min \quad \sum_{(i,j)\in E} x_{ij} - 1$$

$$Min \quad 3 + \sum_{(i,j)\in E} t_{ij}x_{ij} + 5\left(\sum_{(i,j)\in E} x_{ij} - 1\right)$$

$$Min \quad \sum_{(i,j)\in E} p_{ij}x_{ij}$$

$$Max \quad \sum_{(i,j)\in E} f_{ij}x_{ij}$$

$$S.T. \begin{cases} \sum\limits_{(i,j)\in E} x_{ij} - 1 \leq c \\ \sum\limits_{\substack{j=1 \\ (i,j)\in E}}^{n} x_{ij} - \sum\limits_{\substack{j=1 \\ (i,j)\in E}}^{n} x_{ji} = \begin{cases} 1 & (i=s) \\ -1 & (i=e) \\ 0 & (i\neq s,e) \end{cases} \\ x_{ij} \geq 0 \quad (i,j)\in E \end{cases}$$

5 模型II的分析与建立

5.1 公汽、地铁网简化

根据题意，地铁站周围存在多个相邻公汽站点，但并未给出具体位置情况，站点间步行时间难以确定，为使问题得到简化，本文将这些可互换的紧邻站点抽象为一个站点。

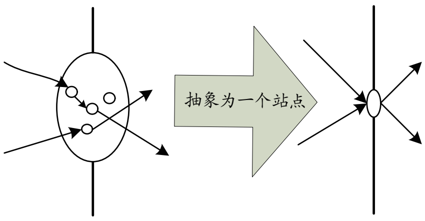

Figure 5: 紧邻站点抽象示意图

在以上假设条件下视地铁同公汽，可以建立公汽、地铁直达数据库Q^D。

5.2 部分目标分析

行程总时间最短：

1）乘车时间： $\sum_{(i,j)\in E} t_{ij}x_{ij}$

2）设$Zij=3$表示$i\to j$最短直达为公汽（也表示始发乘坐公汽等待3分钟）,$Zij=2$为地铁（也表示始发乘坐地铁等待2分钟），总等待时间：

$$T1=\sum_{(i,j)\in E} Z_{ij}x_{ij}$$

3）步行时间：

首先，将相同车型换乘、不同车型换乘的步行时间，一同视为2分钟：

$$T_a=2\left(\sum_{(i,j)\in E} x_{ij}-1\right)$$

出现不同车型换乘，步行时间增加4分钟（图6中虚线表示地铁，空心圆表示地铁站）：

$$T_b=4\sum_{\substack{Z_{ij}=2\\(i,j)\in E}} x_{ij}$$

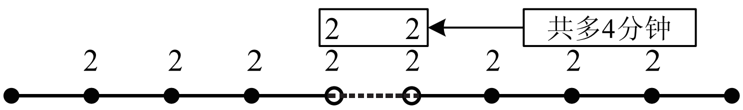

Figure 6: 不同车型换乘示意图

地铁换乘地铁是不同车型换乘的特例，且只可能在D12与D18换乘，出现这种情况在基础上减少步行时间4分钟

$$T_c=4\left(Y^{D12}+Y^{D18}\right)$$

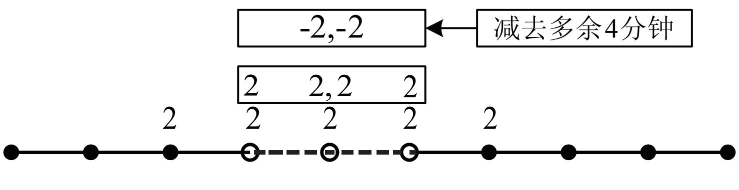

Figure 7: 地铁间换乘示意图

若始发乘坐地铁转公交到达终点，需要增加时间2分钟：

$$T_d=2\sum_{\substack{i=s,j\neq e\\Z_{ij}=2\\(i,j)\in E}} x_{ij}$$

若始发乘坐公交转地铁到达终点，也需要增加时间2分钟：

$$T_e = 2 \sum_{\substack{i \neq s, j=e \\ Z_{ij}=2 \\ (i,j)\in E}} x_{ij}$$

总步行时间$T2 = T_a + T_b - T_c + T_d + T_e$

综上，行程总时间（总等待时间+总步行时间+乘车时间）最短表示为：

$$Min \quad T1 + T2 + \sum_{(i,j)\in E} t_{ij} x_{ij} \tag{7}$$

行程总费用最少：

统计Q^D，建立q_{ij}表示$i \to j$车辆属性：

$$q_{ij} = \begin{cases} 1 & \text{表示单一票制1元} \\ 2 & \text{表示分段计价} \\ 3 & \text{地铁线路} \end{cases}$$

统计Q^D，建立s_{ij}表示$i \to j$过站数目，那么$i \to j$直达费用表示为：

$$p_{ij} = \begin{cases} 1 & q_{ij} = 1 \\ 1 & q_{ij} = 2, s_{ij} \in [1,20] \\ 2 & q_{ij} = 2, s_{ij} \in [21,40] \\ 3 & q_{ij} = 2, s_{ij} \in [41,+\infty] \\ 3 & q_{ij} = 3 \end{cases}$$

行程总费用= 正常票价和− 地铁间换乘减少的费用。

$$Min \quad \sum_{(i,j)\in E} p_{ij} x_{ij} - 3(Y^{D12} + Y^{D18}) \tag{8}$$

5.3 地铁间换乘约束

地铁间换乘只可能出现在站点D12、D18，设变量Y^{D12}、Y^{D18}表示是否地铁间换乘，考虑到模型规模非常大，可构建如下线性约束关联主变量x_{ij}：

$$\sum_{\substack{j=M \\ Z_{ij}=2 \\ (i,j)\in E}} x_{ij} \geq Y^M \quad , \quad \sum_{\substack{i=M \\ Z_{ij}=2 \\ (i,j)\in E}} x_{ij} \geq Y^M \tag{9}$$

$$Y^M \geq 0 \qquad , \quad M = D12, D18$$

考虑到不可能二次换乘地铁，建立如下约束：

$$Y^{D12} + Y^{D18} \leq 1 \tag{10}$$

5.4 模型II的建立

基于前文分析，以式(1)、(7)、(8)、(4)为目标，式(5)、(6)、(9)、(10)为约束，建立线性规划模型如下：

$$Min \quad \sum_{(i,j)\in E} x_{ij} - 1$$

$$Min \quad T1 + T2 + \sum_{(i,j)\in E} t_{ij}x_{ij}$$

$$Min \quad \sum_{(i,j)\in E} p_{ij}x_{ij} - 3(Y^{D12} + Y^{D18})$$

$$Max \quad \sum_{(i,j)\in E} f_{ij}x_{ij}$$

$$S.T. \quad \begin{cases} \sum\limits_{(i,j)\in E} x_{ij} - 1 \le c \\ \sum\limits_{\substack{j=1 \\ (i,j)\in E}}^{n} x_{ij} - \sum\limits_{\substack{j=1 \\ (i,j)\in E}}^{n} x_{ji} = \begin{cases} 1 & (i=s) \\ -1 & (i=e) \\ 0 & (i \ne s,e) \end{cases} \\ x_{ij} \ge 0 \quad (i,j)\in E \\ \sum\limits_{\substack{j=M \\ Z_{ij}=2 \\ (i,j)\in E}} x_{ij} \ge Y^M \\ \sum\limits_{\substack{i=M \\ Z_{ij}=2 \\ (i,j)\in E}} x_{ij} \ge Y^M \\ Y^{D12} + Y^{D18} \le 1 \\ Y^M \ge 0 \quad M = D12, D18 \end{cases}$$

6 模型求解

6.1 邻接算法

邻接算法是标准*Dijkstra*算法的一种扩展，主要是在换乘时需要计算不同的换乘时间与费用；算法中的**距离***根据不同目标，表达式略有不同。

Step1：载入对应权矩阵，输入起、终点s、e；

Step2：若最少换乘矩阵$B_{(se)} = 0$，输出直达线路方案并结束程序；

Step3：若$B_{(se)} = 1$，采用**线路相交法**[5]计算一次换乘方案记录为P；

Step4：初始化$V = \{s\}$，有序链表$U = \{u_1 ... u_n\}$中包含除s外的所有其他顶点，U中顶点按边的权大小升序排列；

Step5：从U中选取第一个顶点k（顶点s到k的边权最小），把k加入V中；

Step6：以k为新考虑的中间点，修改顶点s到U中各顶点的**距离***：若从s到$u(u \in U)$的**距离***比原来小，则修改到达u的**距离***，添加k至路径内；

Step7：出现2次以上转乘方案从s到达e时：与对应P比较，若1次换乘**距离***最短则结束程序，输出P；否则，重复(5)和(6)直到V包含所有顶点。

6.2 线性规划法

文中(4.4)、(5.4)节建立了线性规划模型I、II，本文在$Lingo$软件环境下求解，得到了各目标的全局最优解见(6.3)、(6.4)。

6.3 模型I求解结果

Table 1: 模型I乘车方案表(总耗时包括始发等待时间3分钟)

线路	换乘次数	总耗时（分钟）	转乘站点	乘坐车辆	转乘始发	费用（元）
S3359↓S1828	1*	104	S1784	L436→L167	0	3*
	1	140	S2364	L469→L217	1*	3
	2	67*	S3697→S1784	L484→L485→L167	0	3
S1557↓S0481	2*	109	S1919→S3186	L084→L189→L460	0	3*
	2	130	S1919→S0417	L084→L497→L460	1*	4
	3	102*	S1919→S3186→S903	L084→L189→L091→L239	-	4
S0971↓S0485	1*	131	S2184	L013→L417	0	3*
	2	106*	S2517→S2159	L013→L290→L469	0	3
	2	124	S2324→S2482	L013→L132→L417	1*	4
S0008↓S0073	1*	86	S2263	L355→L345	0	2*
	1	131	S3915	L463→L118	1*	2
	2	70	S1691→S2184	L198→L290→L345	0	3
	4	62*	S3766,S2085,S483,S525	L198,L476,L17,L328,L103	-	5
S0148↓S0485	2*	109	S0036→S2210	L308→L156→L417	1*	3
	3	105*	S3604→S2361→S2210	L308→L81→L156→L417	-	4
S0087↓S3676	1*	68	S3496	L454→L209	0	2*
	2	49*	S0088→S0427	L021→L231→L097	0	3
	2	61	S3874→S0280	L021→L068→L462	1*	3

6.4 模型II求解结果

通过邻接算法与规划算法均可求出各目标全局最优解；下面报表中按字典序分别给出六个实例同时考虑公汽、地铁的乘车方案；其中加*号项表示该项指标最优。

Table 2-1: 模型II乘车方案表(总耗时包括始发等待时间)

线路	换乘次数	总耗时（分钟）	转乘站点	乘坐车辆	转乘始发	费用（元）
S3359↓S1828	1*	104	S1784	L436→L167	0	3*
	1	140	S2364	L469→L217	1*	3
	2	67	S3697→S1784	L484→L485→L167	0	3
	2	79	S1842→S1671	L123→L475→L041	1	3
	4	65*	S2903,D12,D37,S1671	L15,L201,T02,L428,L41	-	7

Table 2-2: 模型II乘车方案表(总耗时包括始发等待时间)

线路	换乘次数	总耗时（分钟）	转 乘 站 点	乘 坐 车 辆	转乘始发	费用（元）
S1557↓S0481	2*	109	D20→S3186	L084→L189→L460	0	3*
	2	121	D20→S1402	L084→L030→L460	1	3
	2	121	D20→S1402	L363→L030→L460	1	3
	3	102*	S1919→S3186→S903	L84→L189→L91→L239	-	4
S0971↓S0485	1*	131	S2184	L013→L417	0	3*
	2	99	D01→D21	L094→T001→L051	0	5
	2	124	S2324→S2482	L013→L132→L417	1*	4
	3	98*	D01→D15→3351	L094→T01→L156→L417	-	6
S0008↓S0073	1*	83	D13	L159→L474	0	2*
	2	58	D30→D25	L150→T002→L103	0	5
	2	63.5	D30→D24	L150→T002→L103	1*	5
	3	56.5*	D15→D12→D25	L200→T01→T02→L103	-	5
S0148↓S0485	2*	90.5	D02→D21	L024→T001→L051	0	5
	2	109	S0036→S2210	L308→L156→L417	1*	3*
	3	89.5*	D02→D15→S3351	L024→T001→L156→L417	-	6
S0087↓S3676	0*	30*	-	T002	-	3
	0	33	-	L231	-	1*
	0	42	-	L381	-	1

7 模型III的分析与建立

本模型主要以出行总时间最省为主要目标，同时考虑转乘次数尽量少，行程费用最少，转乘点始发站最多。有向赋权图中任意两点之间的时间权值有乘车时间t_{ij}^1和步行时间t_{ij}^2，此时对应乘车决策变量为x_{ij}(0-1变量)，步行决策变量为y_{ij} (0-1变量)。

7.1 目标分析

综合考虑步行因素后，只改变了出行总耗时（总等待时间+换乘步行时间+乘车时间+步行时间）：

$$Min \quad T1 + T2 + \sum_{(i,j)\in E} t_{ij}^1 x_{ij} + \sum_{(i,j)\in E} t_{ij}^2 y_{ij} \tag{11}$$

7.2 约束分析

同一线路段只能够在步行与乘车之间选其一，约束表达为：

$$\sum_{\substack{j=1\\(i,j)\in E}}^{n} (x_{ij} + y_{ij}) - \sum_{\substack{j=1\\(i,j)\in E}}^{n} (x_{ji} + y_{ji}) = \begin{cases} 1 & (i = s) \\ -1 & (i = e) \\ 0 & (i \neq s, e) \end{cases} \tag{12}$$

$$x_{ij} + y_{ij} \leq 1 \quad (i,j) \in E \tag{13}$$

7.3 模型III的建立

基于前文分析，以式(11)、(8)、(4)为目标，式(5)、(12)、(13)、(9)、(10)为约束，建立$0-1$整数规划模型如下：

$$Min \quad T1+T2+\sum_{(i,j)\in E} t_{ij}^1 x_{ij}+\sum_{(i,j)\in E} t_{ij}^2 y_{ij}$$

$$Min \quad \sum_{(i,j)\in E} p_{ij}x_{ij}-3(Y^{D12}+Y^{D18})$$

$$Max \quad \sum_{(i,j)\in E} f_{ij}\cdot x_{ij}$$

$$S.T. \quad \begin{cases} \sum\limits_{(i,j)\in E} x_{ij}-1\le c \\ \sum\limits_{\substack{j=1\\(i,j)\in E}}^{n}(x_{ij}+y_{ij})-\sum\limits_{\substack{j=1\\(i,j)\in E}}^{n}(x_{ji}+y_{ji})=\begin{cases}1 & (i=s)\\ -1 & (i=e)\\ 0 & (i\ne s,e)\end{cases} \\ x_{ij},y_{ij}\in\{0,1\} \quad (i,j)\in E \\ x_{ij}+y_{ij}\le 1 \qquad (i,j)\in E \\ \left.\begin{array}{l} \sum\limits_{\substack{j=M\\Z_{ij}=2\\(i,j)\in E}} x_{ij}\ge Y^M \\ \sum\limits_{\substack{i=M\\Z_{ij}=2\\(i,j)\in E}} x_{ij}\ge Y^M \end{array}\right\} M=D12,D18 \\ 0\le Y^{D12}+Y^{D18}\le 1 \end{cases}$$

7.4 模型III算法（略）

8 结果分析、模型的评价与推广（略）

参考文献：

[1] 谢金星、薛毅，《优化建模与LINDO/LINGO软件》，北京，清华大学出版社，2005-7，第一版

[2] Duane Hanselman、Bruce Littlefield，《Matlab 7》，清华大学出版社，2006-5，第一版

[3] 王莉、李文权，《公共交通系统最佳路径算法》，东南大学学报，第34卷：第3期，2004年3月

[4] RICHARD JOHNSONBAUGH、MARCUS SCHAEFER，《ALGORITHMS》，2007年6月

[5] 陈箫枫、蔡秀云、唐德强，《最短路径算法分析及其在公交查询的应用》，工程图学学报，2001年，第3期

[6] 闫小勇、牛学勤，《公交网络多路径选择启发式算法研究》，城市交通，2005年8月，第3卷，第3期

[7] 赵巧霞、马志强、张 发，《以最小换乘次数和站数为目标的公交出行算法》，计算机应用，2004年12月，第24卷，第12期

[8] 徐多勇、李志蜀、梅林，《基于GSM短信息的公交查询系统的最优化转乘方案研究与设计》，计算机应用，2007年6月，第27卷

Best-routing algorithm and model for public transportation systems

TANG ZHi-gao, WANG Ji-li, CAO Ying-ying

Advisor: CAO Hua-lin

(Naval Aeronautical Engineering Academy(Qingdao),Qingdao 266041)

Abstract: In this paper, the information of direct and circle lines is abstracted respectively at first, and cellular array in MATLAB software is used as the carrier to establish a database named Q which stores all the relative information of each direct bus. We build different non-negative directed weighted graph and corresponding weighted matrix referring to different needs of users (the times of bus changing, total time, total expenditure and whether it's a starting station of this bus) . Subsequently, we establish multi-objective linear programming models respectively based on all the analysis carried out above. In order to balancing the space and time complexity of algorithm, Line Intersection *Algorithm*[5] is used to give the feasible solutions when the times of bus changing are less than or equal to two, while Adjacency Algorithm and linear programming method can be adopted when the times are more than two.Finally,integrated solutions are displayed in dictionary sort to the user terminal.

Keywords: Directed weighted graph ,Adjacency algorithm ,Dictionary sort